

Two-dimensional bin packing with one-dimensional resource augmentation

Nikhil Bansal, Maxim Sviridenko*

IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA

Received 17 November 2005; received in revised form 7 September 2006; accepted 14 September 2006

Available online 14 December 2006

Abstract

The two-dimensional bin packing problem is a generalization of the classical bin packing problem and is defined as follows. Given a collection of rectangles specified by their width and height, pack these into a minimum number of square bins of unit size. Recently, the problem was proved to be APX-hard even in the asymptotic case, i.e. when the optimum solutions require a large number of bins [N. Bansal, J. Correa, C. Kenyon, M. Sviridenko, Bin packing in multiple dimensions: Inapproximability results and approximation schemes, *Math. Oper. Res.* 31 (1) (2006) 31–49]. On the positive side, there exists a polynomial time algorithm that uses OPT bins whose sides have length $(1 + \epsilon)$, where OPT denotes the number of unit sized bins used by the optimum solution [N. Bansal, J. Correa, C. Kenyon, M. Sviridenko, Bin packing in multiple dimensions: Inapproximability results and approximation schemes, *Math. Oper. Res.* 31 (1) (2006) 31–49].

A natural question that remains is the approximability of the problem when we are allowed to relax the size of the unit bins in only one dimension. In this paper, we show that there exists an asymptotic polynomial time approximation scheme for packing rectangles into bins of size $1 \times (1 + \epsilon)$.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Bin packing; Approximation scheme

1. Introduction

In the two-dimensional (2D) bin packing problem, we are given a collection of rectangles specified by their width and height that need to be packed into larger square bins. The most interesting and well-studied version of this problem is the so-called *orthogonal packing without rotation* where each rectangle must be packed parallel to the edges of a bin and it cannot be rotated. The goal is to find a feasible packing, i.e. a packing where rectangles do not overlap, using the smallest number of bins. We assume that every rectangle p has width $1 \geq w_p > 0$ and height $1 \geq h_p > 0$. When the height of each rectangle is exactly 1, the problem is easily seen to be identical to the classical one-dimensional bin packing and hence is NP-hard.

The 2D bin packing has many real-world applications: from packing newspaper commercials to cutting-stock problems. Bin packing and its multi-dimensional variations are classic problems in combinatorial optimization and

* Corresponding address: IBM T.J. Watson Research Center, Department of Math Sciences, Yorktown Heights, NY 10598, USA.

E-mail addresses: nikhil@us.ibm.com (N. Bansal), sviri@us.ibm.com (M. Sviridenko).

have been studied extensively. There are surveys devoted to approximation algorithms [4], online algorithms [7], experimental analysis of heuristics and enumerative approaches [11] and average case analysis [5].

We will be interested in polynomial time approximation algorithms in this paper. For packing problems, the worst case approximation ratio usually occurs only for specialized “small” instances, and hence the standard measure used is the asymptotic approximation ratio R^∞ . Given a polynomial time algorithm A , the ratio R_A^∞ is defined by

$$R_A^n = \max\{A(I)/\text{OPT}(I) \mid \text{OPT}(I) = n\}$$

$$R_A^\infty = \lim_{n \rightarrow \infty} \sup R_A^n$$

where I ranges over the set of all problems instances and $A(L)$ (resp. $\text{OPT}(L)$) is the value of the algorithm A (resp. the optimum algorithm) on the instance L . A problem is said to admit an asymptotic approximation scheme (APTAS) if for every $\epsilon > 0$, there is a polynomial time algorithm A_ϵ with an asymptotic approximation ratio of $(1 + \epsilon)$. The running time of the algorithm A_ϵ can have arbitrary dependence on ϵ , but it must have polynomial dependence on the input size of the problem.

In their celebrated work, de la Vega and Lueker [8] gave the first APTAS for the classic 1D bin packing problem. Later, this was substantially improved by Karmarkar and Karp [9] who gave a guarantee of $\text{OPT}(I) + O(\log^2 \text{OPT}(I))$. In contrast, Bansal et al. [1] showed that the 2D case does not admit an APTAS. The best known algorithm for the 2D case is due to Caprara [3] that achieves an asymptotic approximation ratio of $T_\infty \approx 1.69$. Here T_∞ is the well known Harmonic number that appears ubiquitously in bin packing literature. If we relax the requirement that the algorithm uses unit size bins, significantly better results can be obtained. Bansal et al. [1] showed that for any $\epsilon > 0$, there is an algorithm that can pack the rectangles in OPT bins of size $(1 + \epsilon) \times (1 + \epsilon)$, where OPT is the number of unit size bins used by the best possible packing. However, the running time of the algorithm is $O(n^{2^{O(1/\epsilon)}})$ which is rather impractical. Better results are also known of various special cases of 2D bin packing. In particular, an APTAS is known in the case when all the rectangles that need to be packed are squares [1]. An APTAS is also known for the so-called guillotine packing where the rectangles are only allowed to be packed in a certain well-structured way. We refer the reader to [11,2] for further details.

In this paper, we show that there is an asymptotic polynomial time approximation scheme for 2D bin packing even if we allow bins of size $1 \times (1 + \epsilon)$. In particular, given any $\epsilon > 0$, our algorithm finds a packing using $(1 + \epsilon)\text{OPT} + O(1)$ bins of sizes $1 \times (1 + \epsilon)$, where OPT is the number of unit size bins used by the best possible packing. The $O(1)$ term above only depends on ϵ . Unfortunately, at this point our result only seems to be of theoretical interest and the running time of our algorithm is n raised to a tower of exponents of $1/\epsilon$, where n is the number of rectangles in the instance.

Our techniques are similar to those used in most approximation schemes. In particular, given an arbitrary input instance, we round it suitably to obtain a simpler instance with the property that the solution to this simpler instance is closely related to the original one. We also show that the simpler instance has an almost optimum solution with certain structural properties, which allow this solution to be determined in polynomial time. To do this, we will describe several types of transformations on both the instance and the packing in a bin. Since we have slack in the bin size in one dimension only, the rounding techniques need to be relatively sophisticated compared to previously used techniques. We begin by describing some previously known results in Section 2 that we will use later. We describe our structural transformations in Section 3. We then show how to use the structure to obtain an almost optimum packing in Section 4. Finally, in Section 5 we summarize our algorithm and conclude with some open questions in Section 6.

2. Preliminaries

Suppose we are given a collection of rectangles and these need to be packed in a bin of width a and height b . Let us assume that the rectangles are sorted according to non-increasing order of their heights. The Next-Fit-Decreasing-Height shelf heuristic (NFDH) is a packing algorithm that uses the next-fit approach to pack the sorted list of rectangles. The rectangles are packed, left-justified on a shelf (i.e. their base lies on a common horizontal line) until the next rectangle will not fit. This rectangle is used to define a new shelf and the packing continues on this shelf. The earlier shelves are not revisited. If a new shelf does not fit into a current bin we open a new one and place that shelf in a new bin. The earlier bins are not revisited.

In the strip packing problem the rectangles are packed in the strip of width 1 of minimal height. The NFDH algorithm for the strip packing problem works analogously to the NFDH for the 2D bin packing. In the context of strip packing, Coffman et al. [6] showed that

Lemma 1 (Coffman et al. [6], Theorem 1). *Given a collection of two-dimensional rectangles with total area A . The total height (or area) of the strip produced by the NFDH heuristic is at most $2A + 1$.*

Using the same approach it follows that

Lemma 2. *Given a collection of two-dimensional rectangles with total area A . The total number of bins used by the NFDH heuristic is at most $4A + 1$.*

NFDH is particularly useful, if the rectangles to be packed have small dimensions compared to the bin.

Lemma 3 (Coffman et al. [6]). *Given a collection of two-dimensional rectangles. If the NFDH heuristic applied to this set and a bin of size $a \times b$ can not place any other rectangle in the bin, the total wasted (unfilled) volume in that bin is at most $\delta(a + b)$.*

The following lemma bounds the waste produced by the NFDH algorithm it is analogous to Theorem 5 in [6]. The proof of the multi-dimensional version could be found in [1].

Lemma 4. *For any collection of rectangles C , let $A(C)$ denote the total area of rectangles in C . If all rectangles in C have height and width at most δ then all the rectangles can be packed using at most $(1 + 4\delta)A(C) + 1$ unit bins for any $\delta \in [0, 1/4]$.*

3. Structural properties

3.1. Size classification

Let G denote the set of all the rectangles to be packed. Let A denote the total area of the rectangles in G . Let OPT denote the optimum number of unit square bins required to pack all the rectangles in G . As each bin has unit area, clearly OPT is at least A .

Given a fixed $\epsilon > 0$, we define a sequence $\epsilon_0, \epsilon_1, \dots$ recursively as $\epsilon_0 = 1$, $\epsilon_{i+1} = \epsilon 2^{-8/\epsilon^3} \epsilon_i^3$. Consider the sequence of numbers $\epsilon_0, \dots, \epsilon_k$, for $k = 2/\epsilon$. For $i = 0, \dots, k - 1$, let $G_i \subseteq G$ denote the subset of rectangles that have height or width in the interval $(\epsilon_{i+1}, \epsilon_i]$. Since every rectangle lies in at most two sets G_i and there are $k = 2/\epsilon$ sets G_i , there must exist a set G_m such that the total area of the rectangles in G_m is at most ϵA . Let M (for medium) denote the rectangles in G_m . We partition the rectangles in $G \setminus G_m$ as follows:

- Let L (large) denote the rectangles that have both height and width $> \epsilon_m$.
- Let H (horizontal) denote the rectangles with width $> \epsilon_m$ and height $\leq \epsilon_{m+1}$.
- Let V (vertical) denote the rectangles with height $> \epsilon_m$ and width $\leq \epsilon_{m+1}$.
- Let S (small) denote the rectangles with both height and width $\leq \epsilon_{m+1}$.

Consider the instance $G' = G \setminus M$. Given a packing of G' , by Lemma 2 we can extend it to a packing of G using $4\epsilon A + 1 \leq 4\epsilon \cdot \text{OPT} + 1$ additional bins by packing the rectangles in M separately using NFDH. Thus, it suffices to find an almost optimum packing for G' . Observe that in the set G' , any rectangle in S is significantly smaller (in at least one dimension) than any other rectangles in G' . We will adopt the following approach. Let $G'' = G' \setminus S = L \cup H \cup V$. We will find a close to optimum packing for G'' that will additionally have a special structure that will allow us to add the rectangles in S in an “efficient” way to obtain an almost optimum packing for G' . Until Section 4.3 we will focus on finding an optimum structured packing for G'' . Till then, we will abuse the notation somewhat to let OPT denote the optimum number of bins required to pack G'' .

3.2. Fractional packing

We consider a fractional relaxation of the way in which rectangles are allowed to be packed in bins. We call this a *fractional packing*. These ideas were first introduced by Kenyon and Rémila in their breakthrough work on 2D strip packing [10]. In a fractional packing, we relax how rectangles in H and V can be packed in the bins. Each rectangle $R \in H$ can be split horizontally into an arbitrarily many but finite number of rectangles say R_1, \dots, R_m such that for each $1 \leq i \leq m$, $w(R_i) = w(R)$ and $\sum_{i=1}^m h(R_i) = h(R)$. These R_i can then be packed arbitrarily, i.e. they can be placed at non-contiguous locations in a bin or can be placed even in separate bins. Each rectangle $R \in V$ can be split vertically into rectangles R_1, \dots, R_m such that for each $1 \leq i \leq m$, $h(R_i) = h(R)$ and $\sum_{i=1}^m w(R_i) = w(R)$. Again the rectangles R_i can be packed arbitrarily.

Let OPT' denote the number of unit size bins required by the optimum fractional packing of G'' . Clearly $\text{OPT}' \leq \text{OPT}$. We will proceed as follows. We will describe a series of transformations that transform the original instance $I = G''$ into a simpler (but closely related) instance I' with the following properties:

1. Any fractional packing of I that uses B unit size bins implies a packing for I' that uses $(1 + \epsilon)B + f(\epsilon)$ bins of size $1 \times (1 + \epsilon)$, where f is a function of ϵ .
2. Any fractional packing of I' into C bins of size $1 \times (1 + \epsilon)$ implies a fractional packing of I into $(1 + \epsilon)C + g(\epsilon)$ bins of size $1 \times (1 + \epsilon)$, where g is some function of ϵ .

As the instance I' is simpler, we will show how to find a close to optimum fractional packing for I' that in addition has a special structure. To do this, we will use a combination of enumeration and linear programming. By the property above, this gives us an almost optimum fractional packing for I into bins of size $1 \times (1 + \epsilon)$. We then show how to convert the fractional packing for I into an integral one, without adding too many additional bins and still maintaining the structure. Finally, we use this structure to show how rectangles in S can be added while keeping the number of required bins close to optimum.

3.3. Rounding rectangle sizes

We now describe the transformations and show that they maintain the properties mentioned above. The ideas used here are similar in spirit to those introduced by de la Vega and Lueker [8].

Lemma 5. *Consider the instance obtained from I by rounding up the height of each rectangle in V and L to the next integral multiple of $\epsilon\epsilon_m$. Then there exists a feasible fractional packing for the new instance using OPT' bins of size $1 \times (1 + \epsilon)$ with the additional property that each rectangle in V and L has its bottom edge placed at an integral multiple of $\epsilon\epsilon_m$.*

Proof. Consider the optimum fractional packing of I , and focus on a particular bin B . Let C_k (resp. A_k) denote the collection of objects in $L \cup V$ (resp. $L \cup V \cup H$) in the bin B such that their bottom edge has y -coordinate in the interval $[k\epsilon_m, (k+1)\epsilon_m]$. Let $A_{>k}$ be all rectangles in B the bottom edges of which have y -coordinate $> k\epsilon_m$. Note that $A_{>i}$ also contains rectangles in H .

Consider the following procedure applied to the bin B . At step i for $i = 0, \dots, 1/\epsilon_m - 1$, for each rectangle in C_i round its height up to the nearest integral multiple of $\epsilon\epsilon_m$ and shift the rectangle up such by the smallest amount ($< \epsilon\epsilon_m$) such that its bottom edge has y -coordinate an integral multiple of $\epsilon\epsilon_m$. Next shift each rectangle in $A_{>i} \setminus C_i$ up by $2\epsilon\epsilon_m$. Repeat the above procedure for $i = 0, \dots, 1/\epsilon_m - 1$. We show that this procedure gives the desired packing.

Consider step i . The rectangles in C_i move up by at most $\epsilon\epsilon_m$ and their height also increases by at most $\epsilon\epsilon_m$. Since the rectangles in $A_{<i}$ and $A_i \setminus C_i$ do not move at all, these rectangles do not overlap with any rectangle in C_i . Also, as rectangles in $A_{>i} \setminus C_i$ are all moved up exactly by $2\epsilon\epsilon_m$, no rectangle in C_i can overlap with any rectangle in $A_{>i} \setminus C_i$ after the repositioning. Thus no rectangles overlap and the packing produced is feasible at the end of the procedure. Since there are $1/\epsilon_m$ stages and each stage causes a rectangle to move by at most $2\epsilon\epsilon_m$, it follows that the new packing is feasible for a bin of height $1 + 2\epsilon$. ■

We now describe how to round the widths of rectangles in L to a constant number of distinct widths. The idea is to use the well-known rounding for one-dimensional bin packing from [8] for each possible height class.

Lemma 6. *We can round up the widths for rectangles in L such that there are at most $1/(\epsilon\epsilon_m) \cdot 1/(\epsilon\epsilon_m^2)$ distinct widths and the number of bins required increases by a factor of at most $(1 + \epsilon)$.*

Proof. Let L_k , for $k = 1, \dots, 1/\epsilon\epsilon_m$ denote the set of rectangles that have height $k\epsilon\epsilon_m$. By Lemma 5, $\cup_{k=1}^{1/\epsilon\epsilon_m} L_k = L$. For each k , arrange the rectangles in L_k according to the non-increasing order of their widths. We group together every consecutive $\lceil |L_k|/g \rceil$ rectangles, except the last group that may contain less than $\lceil |L_k|/g \rceil$ rectangles, where g is a constant to be specified later. Call these groups, $D_{k,1}, \dots, D_{k,g}$. Next, for each rectangle we round its width to the largest width in its group. Thus in the rounded instance the rectangles in L_k have g distinct widths. As for the one-dimensional bin packing problem any feasible packing of rectangles from L_k before rounding can be used to pack rectangles in $L_k \setminus D_{k,1}$ after rounding, i.e. we could pack rounded rectangles from $D_{k,j}$ for $j = 1, \dots, g$ into the places occupied by the rectangles from $D_{k,j}$ before the rounding since the smallest width in $D_{k,j-1}$ is larger than the largest width in $D_{k,j}$. We assume that for each k for $1 \leq k \leq 1/\epsilon\epsilon_m$, the rectangles in $D_{k,1}$ are packed separately as one rectangle per bin. Thus, we need at most $\sum_{k=1}^{1/\epsilon\epsilon_m} |L_k|/g = |L|/g$ additional bins.

Since any rectangle in L has area at least ϵ_m^2 , the optimum solution uses at least $\epsilon_m^2 |L|$ bins and hence choosing $g = 1/\epsilon\epsilon_m^2$, it follows that the number of bins required does not increase by a factor of more than $(1 + \epsilon)$. ■

We now show how to round up the widths for rectangles in H such that there are a constant number of widths. The technique used in the previous lemma does not apply directly since, unlike in L we do not have a constant number of heights for the rectangles in H . This is where we use the notion of a fractional packing.

Lemma 7. *We can transform the widths of rectangles in H such that there are only $1/(\epsilon\epsilon_m)$ distinct widths and the number of bins required by any fractional packing increases by a factor of at most $(1 + \epsilon)$.*

Proof. Let h' denote the total height of the rectangles in H . We can assume that $h' > 1/\epsilon$, else we trivially pack these rectangles using $O(1/\epsilon)$ additional bins by the NFDH algorithm (Lemma 2). As the total area of rectangles in H is at least $\epsilon_m h'$, any packing of H uses at least $\epsilon_m h'$ bins.

We stack the rectangles in H on top of each other in the order of nondecreasing widths, and make $\frac{1}{\epsilon\epsilon_m}$ groups (possibly splitting some rectangles between two groups) such that each group has an equal height of $\epsilon\epsilon_m h'$. Each rectangle can be in at most 2 groups, since the total height of each group is at least $h'\epsilon\epsilon_m \geq \epsilon_m \geq \epsilon_{m+1}$ and therefore larger than the height of any individual rectangle.

For each rectangle lying in group g_i , we round up its width to that of the rectangle with the least width in group g_{i+1} . Again, since the total height of rectangles in the largest group is $\epsilon\epsilon_m h'$, their area is at most $\epsilon\epsilon_m h' \leq \epsilon \cdot \text{OPT}$. We can pack these separately using Lemma 2.

If we have a feasible fractional solution for the instance where rectangles from H are not rounded we can transform it into a feasible fractional packing for the rounded instance as follows. All items that are not in H stay at the same places in the new packing. All rectangles from the group with largest width are deleted and packed in separate bins using Lemma 2. Rectangles from the group g_i are rounded and moved to the places which were occupied by the rectangles from the group g_{i+1} . As we have a fractional packing, and the cumulative height of rectangles in both groups is the same, the rectangles in g_i can be split and put in places previously occupied by g_{i+1} . The total number of additional bins required is at most $\max\{\epsilon \cdot \text{OPT}, O(1/\epsilon)\}$. ■

3.4. Bin structure: Boxes and cells

We now show that for instances of the type determined by Lemmas 5–7, there exists a close to optimum fractional packing that has a particularly nice structure.

Consider a fractional packing of the rounded instance into $1 \times (1 + \epsilon)$ size bins. For each bin B , we partition it into boxes $B(i, j)$, where the box $B(i, j)$ is the rectangular region determined by $i \cdot \epsilon^2 \epsilon_m^2 \leq x \leq (i + 1) \cdot \epsilon^2 \epsilon_m^2$ and $j \cdot \epsilon\epsilon_m \leq y \leq (j + 1) \cdot \epsilon\epsilon_m$.

Definition 1. A box $B(i, j)$ is *bad* if it overlaps both with some piece of a rectangle in V and some rectangle (or its piece) in $L \cup H$. A box that is not bad is called *good*.

The following lemma gives an important structural property of any feasible packing.

Lemma 8. *In any bin, the number of bad boxes is at most $2(1 + \epsilon)/(\epsilon\epsilon_m^2)$.*

Proof. We first show that if $B(i, j)$ is bad, then either $B(i - k, j)$ for $k = 1, \dots, 1/(\epsilon^2\epsilon_m)$ or $B(i + k, j)$ for $k = 1, \dots, 1/(\epsilon^2\epsilon_m)$ are all good boxes. Thus, for any bad box, either the $\frac{1}{\epsilon^2\epsilon_m}$ boxes immediately on its left or those immediately to its right are good.

Suppose $B(i, j)$ is a bad box, then it overlaps with a piece of some rectangle in V and with a piece R of some rectangle in $(L \cup H)$. Since the width of rectangles in $(L \cup H)$ is at least ϵ_m , R overlaps with at least $\epsilon_m \cdot (1/\epsilon^2\epsilon_m^2) = \frac{1}{\epsilon^2\epsilon_m}$ boxes to the right or to the left of $B(i, j)$. Suppose it overlaps with the boxes on the right. By Lemma 5, all rectangles in V start and end at multiples of $\epsilon\epsilon_m$ in the y -coordinate and have height at least equal to that of a box, it implies that the boxes containing R to the right of B cannot be bad except probably the rightmost box overlapping with R .

To finish the proof of the lemma, suppose that there are more than $2(1 + \epsilon)/(\epsilon\epsilon_m^2)$ bad boxes, then by a counting argument there exists a horizontal row of boxes such that at least $2/\epsilon_m$ of them are bad. Therefore, there exist some i and j such that there are at least 3 bad boxes among the set $B(i, j), B(i + 1, j), \dots, B(i + 1/\epsilon^2\epsilon_m - 1, j)$. However this contradicts that each bad box has at least $1/\epsilon^2\epsilon_m$ good boxes to its immediate left or its immediate right. ■

As there are very few bad boxes, we could remove these and consider packings that only have good boxes. The following lemma makes this precise.

Lemma 9. *Given any fractional packing of the rounded instance, we can transform it into another fractional packing where all the boxes are good and the number of bins required increases by a factor of at most $(1 + O(\epsilon))$.*

Proof. Given any fractional packing, for every bin, remove all the rectangles in V that overlap with a bad box. In the worst case, a rectangle in V has height 1. Thus for any bad box, the total area of rectangles in V that can overlap it is at most the width of a bad box. By Lemma 8, the number of bad boxes in any bin is at most $2/(\epsilon\epsilon_m^2)$ and hence the total area of such rectangles per bin is at most $2(1 + \epsilon)/(\epsilon\epsilon_m^2) \cdot (\epsilon^2\epsilon_m^2) \leq 2(1 + \epsilon)\epsilon$. Packing these rectangles in V separately by Lemma 2, it follows that the number of bins increases by a factor of $1 + 8(1 + \epsilon)\epsilon = 1 + O(\epsilon)$. ■

By Lemma 9 above, we can assume that any box has rectangles only from V or only from $L \cup H$. We now refine the boxes further into smaller rectangular regions that we call *cells*. Each cell will have the property that pieces of rectangles that overlap with it either all lie in V or all in H or all in L . We now show how the cells are formed.

Let X denote the set of all possible different widths for rectangles in L or H . By Lemmas 6 and 7 we know that $|X| \leq 1/(\epsilon^2\epsilon_m^3) + 1/\epsilon_m \leq 2/(\epsilon^2\epsilon_m^3)$. Let P denote all possible numbers in the range $[0, 1]$ that are integer linear combinations of numbers in X and $\epsilon^2\epsilon_m^2$. That is,

$$P = \left\{ k\epsilon^2\epsilon_m^2 + \sum_{i=1}^{|X|} k_i x_i : k, k_i \in \mathbb{Z}_+, x_i \in X, \text{ and } k\epsilon^2\epsilon_m^2 + \sum_{i=1}^{|X|} k_i x_i \leq 1 \right\}.$$

Let $0 = p_1 < p_2 < \dots < p_{|P|} = 1$ denote the points in P . As $|X| \leq 2/(\epsilon^2\epsilon_m^3)$, and as each rectangle in $L \cup H$ has width at least ϵ_m , it follows that each $\sum_i k_i \leq 1/\epsilon_m$ and hence

$$|P| \leq \frac{1}{\epsilon^2\epsilon_m^2} \cdot \left(|X| + 1/\epsilon_m \right) \leq \frac{1}{\epsilon^2\epsilon_m^2} \cdot 2^{|X|+1/\epsilon_m} \leq 2^{3/\epsilon^2\epsilon_m^3}$$

where the last inequality holds for all sufficiently small constants $\epsilon > 0$.

Lemma 10. *Without loss of generality, we can assume that for any rectangle in $H \cup L$, the leftmost point has x -coordinate that lies in P .*

Proof. Consider any feasible fractional packing that satisfies the constraints of Lemma 9. For each bin, consider the rectangles in L and the pieces of rectangles in H (recall that rectangles in H could be packed fractionally), and arrange them in increasing order of the x -coordinate of their leftmost point, breaking ties by the y -coordinate of their lowest point. For each rectangle in $L \cup H$, starting from first rectangle in the order specified above, move it the smallest distance to the left until its leftmost x -coordinate becomes a multiple of $\epsilon^2\epsilon_m^2$ or until it cannot be moved because it is obstructed by another rectangle.

Observe that if the leftmost point of a rectangle was in a box $B(i, j)$, then it stays in the box $B(i, j)$ after this transformation. As no box contains rectangles in both V and $L \cup H$, this implies that rectangles in V never obstruct a rectangle in $L \cup H$ when it is moved to the left. Thus, for every rectangle in $L \cup H$, either its leftmost point has x -coordinate at an integral multiple of $\epsilon^2 \epsilon_m^2$ or its left side touches the right side of another rectangle in $L \cup H$. This implies the desired result. ■

We define a *cell* $C(i, j)$ to be the rectangular region determined by $p_i \leq x \leq p_{i+1}$ and $j \cdot \epsilon \epsilon_m \leq y \leq (j+1) \cdot \epsilon \epsilon_m$. By Lemma 10 it follows that

Lemma 11. *For each cell all the rectangles that overlap with it all lie in H , V or L . Moreover each cell is either empty or totally covered by rectangles.*

The number of cells in each bin is at most $|C| = 1/(\epsilon \epsilon_m) \cdot |P| \leq 2^{4/\epsilon^2 \epsilon_m^3}$ for sufficiently small $\epsilon > 0$.

4. Packing rectangles

4.1. Bin types and packing rectangles in L

By Lemma 11 above, either a cell is empty, or else we can assign a unique label, either H , V or L depending on the rectangles that overlap it. Moreover by Lemmas 5 and 7 we know that there are only a constant number of different heights and widths for rectangles in L . Let q denote the number of different types of rectangles in L , where the type of a rectangle from L is defined by its width and height.

Consider all possible ways to place rectangles in L in a bin. There are only a constant number of such ways (although a huge one) because there are only a constant number of different rectangle types in L , only a constant number (at most $1/\epsilon_m^2$) can be packed in a bin, and there are only a constant number of choices for the leftmost bottom corner of a rectangle. For each such way of packing the bin with L , we label the remaining cells that do not overlap with a rectangle in L , with either H or V or leave it unlabeled. We call this a *bin type*. Let $|T|$ denote the number of different possible bin types. Observe that $|T|$ is a constant. Thus, our algorithm can in time $n^{|T|}$ (hence polynomial time) determine the vector $(n_1, \dots, n_{|T|})$ where n_i is the number of bins of bin type i for $1 \leq i \leq |T|$, used in some optimum solution.

Assuming that n_1, \dots, n_T are known to our algorithm, we now show how to pack H and V (integrally) in these bins without using too many additional bins.

4.2. Packing H and V

We use techniques similar to those used by Kenyon and Rémila to convert their fractional solution into an integral one in the context of strip packing [10]. Consider a cell labeled H : since the height of this cell is much larger than heights of rectangles in H , we can view the packing in this cell as a strip packing. However, the crucial difference from the setting considered by [10] is that we have many cells that need to pack. Thus we also need to determine how to allocate the rectangles in H (resp. V) to the various cells labeled H (resp. V). The idea is to write a linear program that obtains a fractional packing of H and V . We then show how to transform this fractional packing into an actual packing i.e. where the rectangles in H and V are not packed fractionally. We first explain how to pack rectangles from H into cells labeled H . The approach for packing V is similar, but we describe them separately to keep the notation and presentation clearer.

4.2.1. Packing H

For a bin type, consider a maximal union of horizontally consecutive cells labeled by H . This defines a *strip* of some width l and height $\epsilon \epsilon_m$. Consider the collection of all such strips defined by a vector of bin types. Let H_l be the cumulative height of all strips with width l . The total number of different widths for strips is upper bounded by

$$|P|^2 \leq 2^{6/(\epsilon^2 \epsilon_m^3)}.$$

Note that since widths of our cells are different, the width of a strip could be defined by its start and end points and therefore we have at most $|P|^2$ options.

Let $\epsilon_m \leq w_1 \leq \dots \leq w_{|X|}$ be the distinct widths of rectangles in $H \cup L$. Recall that $|X| \leq 2/(\epsilon^2 \epsilon_m^3)$ in our rounded instance.

Definition 2. For every strip of width l we define a set of configurations C_l where each configuration is a vector $(k_1, \dots, k_{|X|})$, such that $\sum_{i=1}^{|X|} k_i w_i \leq l$. That is a configuration is a set of widths of items in H which can be packed together in that strip without overlap in x -coordinate.

Since $\sum_{i=1}^{|X|} k_i \leq 1/\epsilon_m$ we have that $|C_l| = O(1)$.

Let $0 \leq x_{rl} \leq H_l$ be a variable which indicates how much configuration $r \in C_l$ is used cumulatively in a fractional packing of items from H in strips of width l , i.e. every feasible fractional packing of a strip of width l is viewed as consisting of consecutive horizontal slices each containing the same configuration of rectangles. Summing up the height of all such slices with the same configuration in all strips of width l would give us a value of x_{rl} corresponding to a fractional packing.

For every possible width ω for rectangles in H , let h_ω be a cumulative height of rectangles of width ω . Let $a_{\omega r}$ be a number of rectangles of width ω participating in configuration $r \in C_l$. We define the following linear program (LP):

$$\sum_{r \in C_l} x_{rl} = H_l, \quad \text{for all } l, \quad (1)$$

$$\sum_l \sum_{r \in C_l} a_{\omega r} x_{rl} = h_\omega, \quad \text{for all } \omega, \quad (2)$$

$$x_{rl} \geq 0, \quad \text{for all } l, r \in C_l. \quad (3)$$

If there is a feasible fractional packing of rectangles in a given fixed set of bin types we can easily construct a feasible solution of the above LP, by defining a value of variables x_{rl} exactly as described before. Letting (x^*) be a basic feasible solution of that LP we now show how to find a near-optimal integral packing using a small additional number of bins.

The number of constraints of type (1) is $|P|^2$ and the number of constraints of type (2) is $|X|$. By the standard property of basic solutions, the solution (x^*) has at most

$$2^{6/(\epsilon^2 \epsilon_m^3)} + \frac{2}{\epsilon^2 \epsilon_m^3} \leq 2 \cdot 2^{6/(\epsilon^2 \epsilon_m^3)}$$

nonzero variables. Every nonzero variable x_{rl} defines a way to pack a strip of width l using rectangles of widths participating in configuration $r \in C_l$. This gives us a fractional packing of H into our guessed solution. Moreover observe that the unused space in a strip is a rectangle and occurs if and only if the sum of the width of rectangles in a strip is smaller than l .

We now show how to obtain an integral packing of rectangles in H . First for each $x_{rl}^* > 0$ we do the following: Consider a single strip of height x_{rl}^* and start packing arbitrary rectangles from H of widths participating in r placing rectangles of the same width on top of each other. Observe that the amount of wasted area will only be near the top of this strip since we do not allowed rectangles to be packed fractionally now. However, this wasted space will at most be $\epsilon_{m+1} \times l$ as each rectangle in H has height at most ϵ_{m+1} . Thus the total wasted area is at most ϵ_{m+1} times the number of $x_{rl}^* > 0$ which is less than 1 by our choice of ϵ_{m+1} . These rectangles that did not fit in the wasted space can be packed using Lemma 4 in at most 4 additional bins.

We now pack the single strips of height x_{rl}^* into the actual strips allocated for them in the bins. Again, consider the following greedy procedure. Take the strip for $x_{rl}^* > 0$ which was packed with rectangles in H . Cut this strip by horizontal lines that are $\epsilon \epsilon_m$ apart from each other. Throw away rectangles which were cut by those lines and place the remaining rectangles into strips of height $\epsilon \epsilon_m$. As each cell has height $\epsilon \epsilon_m$, this gives a valid packing of rectangles that were not thrown away. Also, note that the total area of rectangles thrown away due to a single line is at most $\epsilon_{m+1} \times l$. Thus the total area thrown per bin is at most $\epsilon_{m+1} \times (1/\epsilon \epsilon_m) \leq \epsilon/4$. These rectangles can be packed in separate bins using Lemma 4. Finally, packing rectangles in H integrally only adds at most $\epsilon/4$ to the unused area in any bins. Thus we have that,

Lemma 12. *All rectangles from H except some set of rectangles of total area $\epsilon \cdot \text{OPT} + O(1)$ can be packed integrally into regions of bins consisting of cells labeled by H . Moreover for any bin, the unused area increases by at most $\epsilon/4$ as compared with the fractional packing. The unused area in the fractional packing consists of a rectangle per cell.*

4.2.2. Packing V

We now pack rectangles from V in the cells marked V using a very similar procedure. However, the main difference from packing H (considered earlier) is that the width of cells marked V could be variable (unlike for cells marked H which all have the same height).

For any bin type, let a maximal collection of vertically consecutive cells labelled with V define a vertical strip. As each cell has height exactly $\epsilon\epsilon_m$, there are $1/(\epsilon\epsilon_m)$ different height types for strips. Let Ω_l denote the cumulative width of all strips of height l in our guessed collection of bins n_1, \dots, n_T . Let $\epsilon_m \leq h_1 \leq \dots \leq h_{|Y|}$ be a sequence of heights of rectangles in W .

Definition 3. For every strip of width l we define a set of configurations C'_l where each configuration is a vector $(k_1, \dots, k_{|Y|})$, such that $\sum_{i=1}^{|Y|} k_i h_i \leq l$. That is, a configuration is a set of heights of items in V which can be packed together in that strip without overlap in y -coordinate.

Since $|Y| \leq 1/(\epsilon\epsilon_m)$ and $\sum_{i=1}^{|X|} k_i \leq 1/\epsilon_m$ we have that $|C'_l| = O(1)$.

Let y_{rl} be the variable which means how much configuration $r \in C'_l$ is used in a fractional packing of items from V in a strip of height l and width Ω_l . For every type of height h of rectangles in V , let w_h be the cumulative width of rectangles of that height. Let a_{hr} be the number of rectangles of height h participating in configuration $r \in C'_l$. We define the following linear program (LP):

$$\sum_{r \in C'_l} y_{rl} = \Omega_l, \quad \text{for all } l, \quad (4)$$

$$\sum_l \sum_{r \in C'_l} a_{hr} y_{rl} = w_h, \quad \text{for all } h, \quad (5)$$

$$y_{rl} \geq 0, \quad \text{for all } l, r \in C'_l. \quad (6)$$

Let (y^*) be some optimum basic feasible solution to this LP. It has at most $2/(\epsilon\epsilon_m)$ nonzero variables. Again as in the previous section we pack items in a few stages.

First for every nonzero y_{rl}^* we greedily pack items into strips of height l and width y_{rl}^* . The items which did not fit in this packing have total area at most $\epsilon_{m+1} \times (\text{number of different strips}) = O(1)$.

Consider each strip of width y_{rl}^* corresponding to a nonzero y_{rl}^* . First concatenate all the strips of width l into a large strip S_l of size $\Omega_l \times l$. Consider the collection of strips V_l in the bins $n_1, \dots, n_{|T|}$, that have height l . We cut this big strip S_l into substrips by vertical lines determined by V_l . The distance between lines is not uniform (it is not a fixed number as in the previous section) since it depends on a width of a corresponding strip of V 's in a bin type. If the item is cut we throw it away and pack the remaining items in corresponding places in bins. We do this for each l . Clearly, the total area of cut items is at most the total number of vertical lines used to cut strips multiplied by ϵ_{m+1} . The number of vertical lines per bin is at most the number of cells in a bin. Hence the total number of vertical lines among all the bins is at most

$$\left(\sum_{i=1}^{|T|} n_i \right) \cdot 2^{4/(\epsilon^2 \epsilon_m^3)} = \text{OPT} \cdot 2^{4/(\epsilon^2 \epsilon_m^3)}.$$

However, by our choice of ϵ_{m+1} , the quantity $\epsilon_{m+1} \cdot 2^{4/(\epsilon^2 \epsilon_m^3)}$ is at most $\epsilon/4$. Thus the unused area added per bin is at most $\epsilon/4$. And the rectangles that need to be packed additionally have area at most $\epsilon \cdot \text{OPT}$. Thus we have that,

Lemma 13. *All rectangles from V except some set of rectangles of total area $\epsilon \cdot \text{OPT} + O(1)$ can be packed integrally into regions of bins consisting of cells labeled by V . Moreover for any bin, the unused area increases by at most $\epsilon/4$ as compared with the unused space in the fractional packing. The unused area in the fractional packing consists of a rectangle per cell.*

4.3. Packing small rectangles

Based on the previous lemmas, we obtain a packing of rectangles in L , H and V that uses at most $(1+\epsilon)\text{OPT} + O(1)$ bins of size $1 \times (1 + \epsilon)$. Moreover, this packing has the property that

Lemma 14. *For a bin B , let $f(B)$ denote the area of the free unused space in this bin. Then, there exists a collection of at most $|C|$ disjoint rectangles in B such that each of these rectangles is unused and the total area of these rectangles is at least $f(B) - \epsilon/2$.*

Proof. By Lemmas 12 and 13 we know that the additional area that may be unused to packing the rectangles integrally is at most $\epsilon/4 + \epsilon/4 = \epsilon/2$. Finally, since the unused area in the fractional packing was rectangular in each cell, and it remains unused after the packing is converted to an integral packing, the desired result follows. ■

We are now ready to prove our main result

Theorem 1. *For every $\epsilon > 0$, there exists a polynomial time algorithm for packing arbitrary rectangles into $(1 + \epsilon) \cdot \text{OPT} + O(1)$, bins of size $(1 + \epsilon) \times 1$, where OPT denotes the number of unit size bins required by the optimum solution.*

Proof. Consider the packing of L , H and V obtained thus far. Consider the bins and the rectangles in S in any arbitrary order and packing them in the rectangular partitions of the free space in the bins using the NFDH heuristic. If no additional bins are used to pack the rectangles in S , the result already follows. If additional bins are required, then we claim that each bin (except possibly one) has at most ϵ unused area. Again this will imply the result directly by the area bound. To see the claim, by Lemma 3 for any rectangular region of size $a \times b$, packing items in S into this region using NFDH leads to at most $\epsilon_{m+1}(a + b)$ waste. As $a + b \leq 2$, this waste is at most $2\epsilon_{m+1}$ for any region. By Lemma 14, there are at most $|C|$ rectangles considered for packing S in any bin. Thus the space wasted per bin in these $|C|$ rectangles is at most $2\epsilon_{m+1} \cdot |C| \leq \epsilon/2$, by our choice of ϵ_{m+1} and the upper bound of $|C|$. Again, by Lemma 14, the additional unused area that does not lie in these $|C|$ rectangles is at most $\epsilon/2$. Thus, the total unused area in any bin is at most ϵ . This gives us the desired result. ■

5. Summary of the algorithm

In this section we summarize all steps of our asymptotic polynomial time approximation scheme.

First (in Section 3) we define five classes of rectangles according to their size: Large (L), Vertical (V), Horizontal (H), Medium (M) and Small (S). The total area of the medium rectangles is negligible and therefore they could be packed separately by NFDH. This gives us a large separation in size between items in S and those not in S , and allows us to focus on obtaining a packing of $L \cup H \cup V$.

In the second step, we round up sizes of items in H , V and L and prove that this rounded instance has a feasible fractional solution close to the optimal solution for the original instance. To do this rounding, first we round the heights of items in L and V (Lemma 5). Then we round widths of items in L (Lemma 6). Finally we round widths of rectangles in H (Lemma 7). We then show (described below) how to find an almost optimum fractional solution to this rounded instance.

Based on the sizes of rounded items we define cells, i.e. subdivisions of each bin. In particular, each bin is divided into a constant number of cells and this subdivision is identical for each bin (and hence can be performed by our algorithm without any knowledge of the optimum solution). We then show (Lemma 11) that there is an almost optimum fractional solution that is “well structured”, in the sense that each cell only overlaps with items of a single type (i.e. one of either L , V or H). This allows us to define a constant number of different possible bin types, where each bin type determines for each cell what type of large items should be placed in it and allocates remaining cells either for items in H or in V . Since there is an almost optimum fractional packing that uses only these bin types, our algorithm guesses (by brute force enumeration) how many bin types of each type are used in some optimum solution (Section 4.1). Since there are only a constant number of different bin types, there are at most polynomially many choices to enumerate.

Having guessed the right number of bins of each bin type, we convert this almost optimum fractional packing into an integral packing by solving a linear program for packing items in H and V (Sections 4.2.1 and 4.2.2). We use the

“well-structured” property to argue that the fractional to integral conversion can be done without requiring too many additional bins (Lemmas 12 and 13). In the final step, items from S are packed using NFDH in the areas left empty by the previous packing. This is done in a (standard) way that keeps the packing almost optimum (Lemma 14).

6. Open questions

A natural question is whether the running time dependence on $1/\epsilon$ of our algorithm can be improved. The running time is a tower of exponents of length proportional to $1/\epsilon$, which makes it quite impractical in practice. As a first step in this direction, it would be interesting to reduce the running time dependence in the case when resource augmentation is allowed in both dimensions.

For the d -dimensional case for $d > 2$, it is not known where there is an APTAS even if the algorithm is allowed resource augmentation in d dimensions. The APX hardness for two-dimensional bin packing rules out the possibility of an APTAS for d -dimensional bin packing even when resource augmentation is allowed in $d - 2$ dimensions. It is possible that resource augmentation in $d - 1$ dimensions might be necessary and sufficient for obtaining an APTAS in the d -dimensional case. Settling this would be very interesting.

References

- [1] N. Bansal, J. Correa, C. Kenyon, M. Sviridenko, Bin packing in multiple dimensions: Inapproximability results and approximation schemes, *Math. Oper. Res.* 31 (1) (2006) 31–49.
- [2] N. Bansal, A. Lodi, M. Sviridenko, A tale of two dimensional bin packing, in: *Proceedings of FOCS, 2005*, pp. 657–666.
- [3] A. Caprara, Packing two-dimensional bins in harmony, in: *Proceeding of the 43rd IEEE Symposium on Foundations of Computer Science, FOCS, 2002*, pp. 490–499.
- [4] E. Coffman, M. Garey, D. Johnson, Approximation algorithms for bin packing: A survey, in: D. Hochbaum (Ed.), *Approximation Algorithms for NP-Hard Problems*, PWS Publishing, Boston, 1996, pp. 46–93.
- [5] E.G. Coffman, C. Courcoubetis, M.R. Garey, D.S. Johnson, P.W. Shor, R.R. Weber, M. Yannakakis, Bin packing with discrete item sizes, Part i: Perfect packing theorems and the average case behavior of optimal packings, *SIAM J. Discrete Math.* 13 (2000) 384–402.
- [6] E.G. Coffman, M.R. Garey, D.S. Johnson, R.E. Tarjan, Performance bounds for level-oriented two dimensional packing algorithms, *SIAM J. Comput.* 9 (1980) 808–826.
- [7] J. Csirik, G.J. Woeginger, On-line packing and covering problems, in: A. Fiat, G. Woeginger (Eds.), *Online Algorithms: The State of the Art, 1998*, pp. 147–177.
- [8] W.F. de la Vega, G. Lueker, Bin packing can be solved within $1 + \epsilon$ in linear time, *Combinatorica* 1 (1981) 349–355.
- [9] N. Karmarkar, R.M. Karp, An efficient approximation scheme for the one-dimensional bin-packing problem, in: *Foundations of Computer Science, FOCS, 1982*, pp. 312–320.
- [10] C. Kenyon, E. Rémila, A near-optimal solution to a two-dimensional cutting stock problem, *Math. Oper. Res.* 25 (2000) 645–656.
- [11] A. Lodi, S. Martello, M. Monaci, Two-dimensional packing problems: A survey, *European J. Oper. Res.* 141 (2002) 241–252.